**EJERCICIO1**

**Parte 1: Iniciar sesión desde SQL Server Managment Studio**

**Parte 2: Creación de la base de datos, esquemas y tablas**

1. Crear la base de datos

Crea una BD de nombre alxx

(xx es el número de puesto en clase 01,02……11)

CREATE DATABASE profesor

2. Crear dos esquemas en la base de datos

Uno de ellos llamado alumno y otro nota

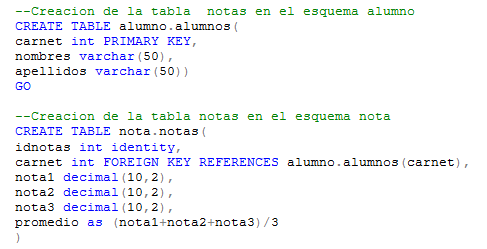
CREATE SCHEMA alumno;

Go

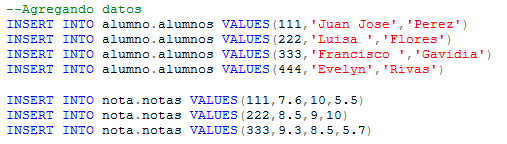
CREATE SCHEMA nota;

Go

3. Crear las tablas de la base de datos, cada una en un esquema diferente



4. Agregar datos a la base de datos



5. Verifica haciendo una consulta de selección que los datos está bien insertados. Nombra correctamente a los objetos.

SELECT \* FROM alumno.alumnos

SELECT \* FROM nota.notas

**Parte 3: Creación de inicio de sesión con SQL Management Studio y Transact SQL**

1. En SQL Server Management Studio crear un inicio de sesión con autenticación en el servidor, con las siguientes características:
   1. Nombre de inicio de sesión: inicioalxx
   2. Pasword: Colegio01
   3. No marcar exigir política de contraseñas
   4. Base de datos predeterminada: alxx
2. En el Editor de consultas, mediante comandos Transact-SQL cree un inicio de sesión, con las siguientes características:
   1. Nombre de inicio de sesión: cominicioalxx
   2. Pasword: 12345

CREATE LOGIN cominicioxx

WITH PASSWORD ='12345';

1. Verifique que se han creado los nuevos inicios de sesión

**Parte 4. Creación de usuarios de base de datos**

1. Conectarse al servidor de SQL Server para colocar el inicio de sesión que acabamos de crear cominicioalxx. ¿Qué sucede?. Soluciona el problema mediante instrucción T-SQL. (Nota: **esquema predeterminado** (Default schema) **alumno)**

USE profesor;

GO

CREATE USER pepe FOR LOGIN cominicioxx with default\_schema=alumno

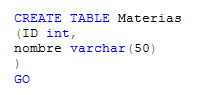
GO

1. Ahora nos logueamos con el inicio de sesión que hemos creamos para que se active el usuario de base de datos.
2. Hacer doble clic en cualquier base de datos, diferente a la de alxx, ¿Qué sucede?
3. Expandimos la base de datos **alxx** y verificamos a que tablas tenemos acceso
4. ¿Qué pasa si quisiéramos acceder a la tabla Notas con una instrucción SELECT?

**Parte 5. Asignación de permisos**

1. Regresar al inicio de sesión de autenticación por Windows (el que se estableció al instalar SQL Server)

2. En la base de datos **alxx**, y crear la siguiente tabla:



3. Agregar los siguientes registros

INSERT INTO Materias VALUES(101,'Fisica')

INSERT INTO Materias VALUES(102,'Programacion')

INSERT INTO Materias VALUES(103,'Diseño Web') INSERT INTO Materias VALUES(104,'Algebra')

4. ¿Bajo qué esquema se queda la tabla Materias?

Bajo el esquema dbo.

5. Ahora en la carpeta seguridad de la base de datos alxx, dar doble clic sobre el usuario de base de datos que se creó anteriormente **(inicioalxx**), y en la sección de asegurables le damos a buscar, apareciendo las siguientes opciones:

**Objetos específicos (Specific objects…):** crearemos permisos para uno o varios objetos de SQL, el usuario tiene la oportunidad de agregar los objetos de su preferencia.

**Todos los objetos de los tipos (All objects of the types…)**: se seleccionaran todos los tipos de objetos, es decir si selecciona el objeto tabla, se seleccionaran todas las tablas que contenga la BD

**Todos los objetos que pertenecen al esquema (All objects belonging to the schema):** podrá seleccionar los objetos que pertenezcan a un esquema en específico.

1. Hay que asignar ciertos permisos a la tabla que acabamos de crear, como solo queremos una tabla seleccionamos la primera opción, ***objetos específicos (Specific Objects)***, en la ventana damos clic en Tipos de objeto (Object Types…)y seleccionamos ***tablas (Tables)***. Buscamos dentro de las las tablas existentes en la base de datos la tabla Materiales. Establecemos que el usuario solo pueda hacer consultas y actualizar datos.
2. Realizar lo mismo que en el punto 6 , pero para el usuario **cominicioalxx** mediante comandos T-SQL, es decir que el usuario solo pueda actualizar datos y realizar consultas en la tabla Materiales.

GRANT SELECT, UPDATE ON dbo.Materias TO pepe

1. Conéctate de nuevo al servidor e ingresa con uno de los dos inicio de sesión que se han creado anteriormente para ver/probar los permisos en esa tabla. ( NOTA: también se puede usar la instrucción EXECUTE AS USER =’cominicioalxx’).

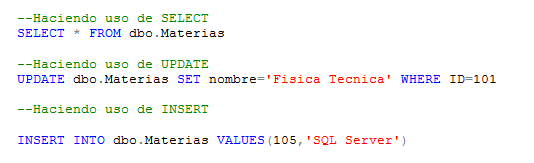
Prueba las siguientes sentencias

EXECUTE AS USER ='pepe'

SELECT \* FROM dbo.Materias

UPDATE dbo.Materias SET nombre='Física' WHERE ID=101;

INSERT INTO dbo.Materias values (105,'sql server')



¿Las tres consultas dieron resultados correctos sí o no?

¿Por qué? La última no la puedo ejecutar ya que no tengo permisos

**Parte 6. Agregando permisos a esquemas**

En los ejercicios anteriores, un usuario de base de datos podía realizar consultas y actualizaciones de datos, ahora haremos que ese usuario de base de datos tenga el permiso para crear tablas las cuales quedaran en el esquema que se le predetermino cuando se estaba creando.

1. Regresar al inicio de sesión de autenticación por Windows (el que se estableció al instalar SQL Server)
2. Para la base de datos alxx , vamos a otorgar permisos al usuario **inicioalxx,** sobre el esquema alumno (seleccionar esquema **alumno** y dar clic en propiedades (Properties))
3. Conceder al usuario inicioalxx permisos para crear tablas y procedimientos dentro del esquema alumno

(En la parte izquierda dar clic en la página de permisos (Permissions), y luego dar clic en la parte donde dice **Ver permisos de base de datos (View database permissions)**. En la sección de permisos explícitos que se encuentra en la parte de abajo, seleccionar los permisos que deseamos que posea el usuario, de clic en la casilla conceder a la par del permiso **Create procedure y Create table)**

1. Realiza lo mismo que en el anterior punto 3 pero mediante comando de T-SQL y para el usuario cominicioalxx.

use profesor;

GRANT ALTER ON Schema :: alumno TO pepe

GRANT CREATE TABLE TO pepe

Grant CREATE procedure to pepe;

GO

1. Probar los permisos, ingresar a SQL Server con el login inicioalxx, ejecute las siguientes instrucciones en la ventana de consultas.

Crear una tabla

USE Alumnos\_Carnet

GO

CREATE TABLE Prueba

(column1 INT NOT NULL, column2 CHAR(10) NOT NULL

)

¿Pudo crearlo, por qué? No porque no tengo permisos sobre el esquema dbo. Si creo la tabla sobre alumno.prueba si puedo

execute as user='pepe'

USE profesor

GO

CREATE TABLE alumno.Prueba

(column1 INT NOT NULL, column2 CHAR(10) NOT NULL

)

Crear una vista

CREATE VIEW Reporte as

SELECT carnet,nombres,apellidos FROM alumno.alumnos

¿Pudo crearlo, por qué? No tengo permisos crear vista.

1. Regresar al inicio de sesión de autenticación por Windows (el que se estableció al instalar SQL Server) y otorgue mediante comandos permiso al usuario inicioalxx para crear vistas.

USE profesor;

GRANT CREATE VIEW TO pepe

1. Cree de nuevo la vista con el inicioalxx que creo en la práctica ¿Se pudo crear sí, no y porque?

execute as user='pepe'

USE profesor

GO

CREATE VIEW Reporte as

SELECT column1 FROM alumno.prueba

No me deja crearla. Es porque estaba intentando crear la vista sobre el esquema dbo (por defecto). Si pongo alumno.Reporte si me deja.

execute as user='pepe'

USE profesor

GO

CREATE VIEW alumno.Reporte as

SELECT column1 FROM alumno.prueba

**Parte 7. Eliminando inicio de sesión**

1. Utilizando el MSSMS:

a. Expanda la carpeta seguridad

b. Busque el inicio de sesión que desea eliminar c. Haga clic derecho

d. Seleccione eliminar

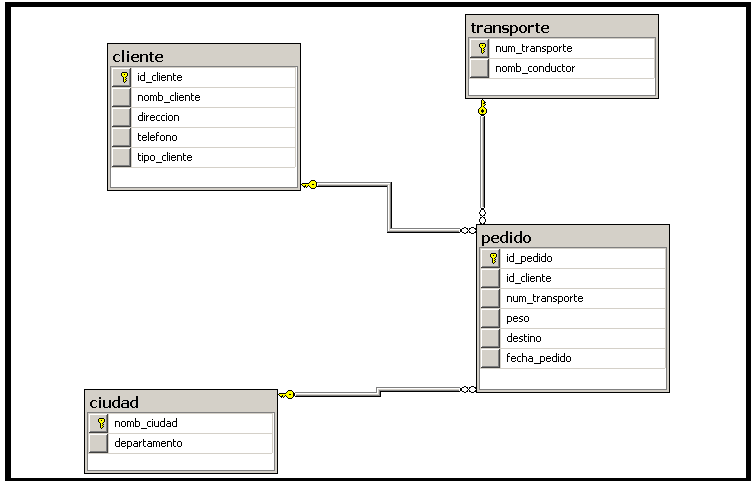
2. Utilizando Transact-SQL

Nota: utilice cualquiera de las dos opciones anteriores para eliminar el login **cominicioalxx**

**EJERCICIO2**

1. Crear la siguiente base de datos mediante comandos.

CREATE DATABASE ejerdos



USE ejerdos;

go

if object\_id('cliente')is not null

drop table cliente;

CREATE TABLE cliente(

id\_cliente int Primary KEY,

nomb\_cliente varchar(30),

direccion varchar(30),

telefono varchar (9),

tipo\_cliente varchar(20)

);

if object\_id('ciudad')is not null

drop table ciudad;

CREATE TABLE ciudad(

nomb\_ciudad varchar(20) Primary Key,

departamento varchar(20)

);

if object\_id('transporte')is not null

drop table transporte;

CREATE TABLE transporte(

num\_transporte int primary key,

nom\_conductor varchar(30)

);

if object\_id('pedido')is not null

drop table pedido;

CREATE TABLE pedido(

id\_pedido int primary key,

id\_cliente int,

num\_transporte int,

peso int,

destino varchar(20),

fecha\_pedido datetime

);

ALTER TABLE pedido ADD CONSTRAINT FK\_pedido\_cliente FOREIGN KEY

(id\_cliente) REFERENCES cliente (id\_cliente);

ALTER TABLE pedido ADD CONSTRAINT FK\_pedido\_transporte FOREIGN KEY

(num\_transporte) REFERENCES transporte (num\_transporte);

ALTER TABLE pedido ADD CONSTRAINT FK\_pedido\_ciudad FOREIGN KEY

(destino) REFERENCES ciudad (nomb\_ciudad);

2. Crear los siguientes esquemas y colocar en ellos las siguientes tablas:

a. Esquema: Datos

i. Tablas: Clientes y ciudades b. Esquema: Pedidos

i. Tablas: Pedido y Transporte

USE ejerdos;

go

CREATE SCHEMA Datos;

go

CREATE SCHEMA Pedidos;

ALTER SCHEMA Datos TRANSFER dbo.cliente;

ALTER SCHEMA Datos TRANSFER dbo.ciudad;

ALTER SCHEMA Pedidos TRANSFER dbo.pedido;

ALTER SCHEMA Pedidos TRANSFER dbo.transporte;

3. Para los usuarios de la Base de Datos se crearán los siguientes tipos de usuarios:

a. Crear un usuario el cual se encarga de ingresar nuevos clientes y ciudades, también a este usuario se le concede el permiso de actualizar y eliminar información de los clientes y ciudades y solo podrá ver esas tablas nada más. Lo creo como administrador y en consulta general

create login user1 with password ='Colegio01';

go

Create user user1 for login user1 WITH default\_Schema=Datos;

USE ejerdos;

go

GRANT SELECT, UPDATE, INSERT, DELETE ON SCHEMA::Datos TO user1;

Concedo los permisos a nivel de esquema para que pueda acceder a las dos tablas que hay en ese esquema y no tenga que ir dando permisos tabla a tabla.

use ejerdos

go

execute as user= 'user1';

INSERT INTO Datos.ciudad values('Madrid', 'Compras'),('Logroño','ventas')

INSERT INTO Datos.cliente values(1, 'pepe','Huesca,32',null,null),(2,'Juan','Duques Nájera',null,null)

use ejerdos

go

execute as user= 'user1';

SELECT \* FROM Datos.ciudad;

SELECT \*FROM Datos.cliente;

use ejerdos

go

execute as user= 'user1';

UPDATE Datos.ciudad set departamento='barcelona'

Where nomb\_ciudad='Logroño'

b. Luego una persona encargada de seleccionar los pedidos que se han realizado y este no podrá seleccionar las otras tablas o información.

create login user2 with password ='Colegio01';

go

Create user user2 for login user2 WITH default\_Schema=Pedidos;

USE ejerdos;

GRANT SELECT ON Pedidos.pedido TO user2;

c. Un usuario administrativo el cual se encarga de seleccionar, agregar, modificar y eliminar

datos de las tablas Pedido y transporte, y solamente verá estas 2 tablas.

create login user3 with password ='Colegio01';

go

Create user user3 for login user3 WITH default\_Schema=Pedidos;

USE ejerdos;

go

GRANT SELECT, UPDATE, INSERT, DELETE ON SCHEMA::Pedidos TO user3;